Marketing Funnels for Calculating Conversion Rates

Gabriel Pena

(Fictional Data / Codecademy and Warby Parker collaboration project)

This is the code for a quiz funnel.
The survey table will contain 5 questions that users answer or do not answer.
We've limited the query to only display 10 rows of users.

```
2   SELECT *
3   FROM survey
4   LIMIT 10;
```

Query Results

question	user_id	response
1. What are you looking for?	005e7f99-d48c-4fce-b605-10506c85aaf7	Women's Styles
2. What's your fit?	005e7f99-d48c-4fce-b605-10506c85aaf7	Medium
3. Which shapes do you like?	00a556ed-f13e-4c67-8704-27e3573684cd	Round
4. Which colors do you like?	00a556ed-f13e-4c67-8704-27e3573684cd	Two-Tone
1. What are you looking for?	00a556ed-f13e-4c67-8704-27e3573684cd	I'm not sure. Let's skip it.
2. What's your fit?	00a556ed-f13e-4c67-8704-27e3573684cd	Narrow
5. When was your last eye exam?	00a556ed-f13e-4c67-8704-27e3573684cd	<1 Year
3. Which shapes do you like?	00bf9d63-0999-43a3-9e5b-9c372e6890d2	Square
5. When was your last eye exam?	00bf9d63-0999-43a3-9e5b-9c372e6890d2	<1 Year
2. What's your fit?	00bf9d63-0999-43a3-9e5b-9c372e6890d2	Medium

Using this code we can count for how many users went on from qst 1 to qst 2, then qst 3, and so on, because many users will give up at random points.
The code above is for another quiz funnel except we will now aggregate the ‘user_id’ column and use a GROUP BY command.
This will count every users for every specific question that is on the survey.

```
7  SELECT question,
8    COUNT(DISTINCT user_id)
9   FROM survey
10  GROUP BY 1;
```

question	COUNT(DISTINCT user_id)
1. What are you looking for?	500
2. What's your fit?	475
3. Which shapes do you like?	380
4. Which colors do you like?	361
5. When was your last eye exam?	270

the results for qst 2 has a 95% / qst 3 has an 80% / qst 4 has a 95% / qst 5 has a 74%

question	COUNT(DISTINCT user_id)
1. What are you looking for?	500
2. What's your fit?	475
3. Which shapes do you like?	380
4. Which colors do you like?	361
5. When was your last eye exam?	270

Using the chart from the previous query, we can calculate the completion rate per question by the number of users for each one. These results can be inserted in a spreadsheet for calculations.

Here we are first getting familiar with the values that each table contains, which is why we only limit the query to 5 rows each so we are not overwhelmed with so much data.
The Warby Parker purchase funnel is Take the Style Quiz → Home Try-On → Purchase the Perfect Pair of Glasses.
For this we are using: Quiz table / Home_try_on table / Purchase table

```
16  SELECT *
17  FROM quiz
18  LIMIT 5;
19
20  SELECT *
21  FROM home_try_on
22  LIMIT 5;
23
24  SELECT *
25  FROM purchase
26  LIMIT 5;
```

user_id	style	fit	shape	color
4e8118dc-bb3d-49bf-85fc-cca8d83232ac	Women's Styles	Medium	Rectangular	Tortoise
291f1cca-e507-48be-b063-002b14906468	Women's Styles	Narrow	Round	Black
75122300-0736-4087-b6d8-c0c5373a1a04	Women's Styles	Wide	Rectangular	Two-Tone
75bc6ebd-40cd-4e1d-a301-27ddd93b12e2	Women's Styles	Narrow	Square	Two-Tone
ce965c4d-7a2b-4db6-9847-601747fa7812	Women's Styles	Wide	Rectangular	Black

user_id	number_of_pairs	address
d8addd87-3217-4429-9a01-d56d68111da7	5 pairs	145 New York 9a
f52b07c8-abe4-4f4a-9d39-ba9fc9a184cc	5 pairs	383 Madison Ave
8ba0d2d5-1a31-403e-9fa5-79540f8477f9	5 pairs	287 Pell St
4e71850e-8bbf-4e6b-accc-49a7bb46c586	3 pairs	347 Madison Square N
3bc8f97f-2336-4dab-bd86-e391609dab97	5 pairs	182 Cornelia St

user_id	product_id	style	model_name	color
00a9dd17-36c8-430c-9d76-df49d4197dcf	8	Women's Styles	Lucy	Jet Black
00e15fe0-c86f-4818-9c63-3422211baa97	7	Women's Styles	Lucy	Elderflower Crystal
017506f7-aba1-4b9d-8b7b-f4426e71b8ca	4	Men's Styles	Dawes	Jet Black
0176bfb3-9c51-4b1c-b593-87edab3c54cb	10	Women's Styles	Eugene Narrow	Rosewood Tortoise
01fdf106-f73c-4d3f-a036-2f3e2ab1ce06	8	Women's Styles	Lucy	Jet Black

Lets first note that the home_try_on table contains an A/B test that was conducted for users to try on either 3 pairs of glasses or 5 pairs.

```
29  SELECT DISTINCT q.user_id,
30                  h.user_id IS NOT NULL AS
    'is_home_try_on',
31                  h.number_of_pairs,
32                  p.user_id IS NOT NULL AS
    'is_purhase'
33    FROM quiz AS q
34    LEFT JOIN home_try_on AS h
35      ON q.user_id = h.user_id
36    LEFT JOIN purchase AS p
37      ON p.user_id = q.user_id
38  LIMIT 10;
```

This query will help us return the results of what the outcome was from trying on either 3 pairs or 5 pairs using all 3 tables.
This query though will only return 10 random users so that our query won't crash with the amount of users there are is in the database.
We are selecting the user_id column from quiz table, home_try_on table, and purchase table. If the user_id has any value in the home_try_on table then it is True (it'll include a 1 on the new column 'is_home_try_on'), If the user_id has a value in purchase table then it is True (it'll include a 1 in the new column 'is_purchase'), number_of_pairs column from home_try_on
For all values to be queried, all table have to be combined using the LEFT JOIN to combine the quiz funnel down to the purchase funnel.
With the LEFT join we will match the user_id column from each table and rename each table:
Quiz is q
home _try_on is h
Purchase is p

```sql
SELECT DISTINCT q.user_id,
                h.user_id IS NOT NULL AS
  'is_home_try_on',
                h.number_of_pairs,
                p.user_id IS NOT NULL AS
  'is_purhase'
  FROM quiz AS q
  LEFT JOIN home_try_on AS h
    ON q.user_id = h.user_id
  LEFT JOIN purchase AS p
    ON p.user_id = q.user_id
LIMIT 10;
```

Now that we have explained what this query will do, lets review the results that have been queried.

We can see which users participated in the ‘home_try_on’ where there is a 1 on the renamed column ‘is_home_try_on’, how many pairs each user was a part of trying, and if the users made the purchase after the try-on, where there is a 1 in the renamed column ‘is_purchase’.

user_id	is_home_try_on	number_of_pairs	is_purhase
4e8118dc-bb3d-49bf-85fc-cca8d83232ac	1	3 pairs	0
291f1cca-e507-48be-b063-002b14906468	1	3 pairs	1
75122300-0736-4087-b6d8-c0c5373a1a04	0	Ø	0
75bc6ebd-40cd-4e1d-a301-27ddd93b12e2	1	5 pairs	0
ce965c4d-7a2b-4db6-9847-601747fa7812	1	3 pairs	1
28867d12-27a6-4e6a-a5fb-8bb5440117ae	1	5 pairs	1
5a7a7e13-fbcf-46e4-9093-79799649d6c5	0	Ø	0
0143cb8b-bb81-4916-9750-ce956c9f9bd9	0	Ø	0
a4ccc1b3-cbb6-449c-b7a5-03af42c97433	1	5 pairs	0
b1dded76-cd60-4222-82cb-f6d464104298	1	3 pairs	0

```sql
WITH funnels AS
(SELECT DISTINCT q.user_id,
                h.user_id IS NOT NULL AS
   'is_home_try_on',
                h.number_of_pairs,
                p.user_id IS NOT NULL AS
   'is_purchase'
  FROM quiz AS q
  LEFT JOIN home_try_on AS h
    ON q.user_id = h.user_id
  LEFT JOIN purchase AS p
    ON p.user_id = q.user_id)

SELECT number_of_pairs,
          COUNT (*) AS 'num_browsers',
          SUM (is_home_try_on) AS
'try_on_users',
          SUM (is_purchase) AS
'user_purchesers',
          1.0 * SUM (is_home_try_on) /
          COUNT (user_id) AS 'browse_to_try_on',
          1.0 * SUM (is_purchase) / SUM
(is_home_try_on) AS 'try_on_to_purchase'
FROM funnels
GROUP BY number_of_pairs;
```

Now that we have the funnels table set up we can calculate and compare the date in several ways. Here we are calculating the percentage of browsers who made a purchase based on what a/b quiz they were a part of, the 3 pairs to try, or the 5 pairs to try. We do this by adding the ‘number_of_pairs’ column at the beginning and aggregating the next 5 columns in regards to the values in the the first column, ‘number_of_pairs’.

number_of_pairs	num_browsers	try_on_users	user_purchesers	browse_to_try_on	try_on_to_purchase
Ø	250	0	0	0.0	Ø
3 pairs	379	379	201	1.0	0.530343007915567
5 pairs	371	371	294	1.0	0.792452830188679